



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/503,215	02/12/2000	Jack Niewiadomski	777.364US1	6252

7590 07/01/2003

John E Whitaker
Merchant & Gould PC
P O Box 2903
Minneapolis, MN 55402-0903

EXAMINER

NAHAR, QAMRUN

ART UNIT

PAPER NUMBER

2124

DATE MAILED: 07/01/2003

9

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary**Application No.**

09/503,215

Applicant(s)

NIEWIADOMSKI ET AL.

Examiner

Qamrun Nahar

Art Unit

2124

— The MAILING DATE of this communication appears on the cover sheet with the correspondence address —
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 16 April 2003.

2a) This action is FINAL. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-26,28-33,36 and 41-47 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) Claim(s) _____ is/are allowed.

6) Claim(s) 1-26,28-33,36 and 41-47 is/are rejected.

7) Claim(s) _____ is/are objected to.

8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11) The proposed drawing correction filed on _____ is: a) approved b) disapproved by the Examiner.
If approved, corrected drawings are required in reply to this Office action.

12) The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) All b) Some * c) None of:
1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
* See the attached detailed Office action for a list of the certified copies not received.

14) Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).
a) The translation of the foreign language provisional application has been received.

15) Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) Notice of References Cited (PTO-892)
2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____

4) Interview Summary (PTO-413) Paper No(s). _____
5) Notice of Informal Patent Application (PTO-152)
6) Other: _____

DETAILED ACTION

1. This action is in response to the amendment filed on 4/16/03.
2. The objection to the declaration is withdrawn in view of applicant's remarks.
3. The objections to the abstract are withdrawn in view of applicant's amendment.
4. The objection to the specification is withdrawn in view of applicant's remarks.
5. The rejections under 35 U.S.C. 112, second paragraph to claims 9-10, 21 and 22 are withdrawn in view of applicant's amendment.
6. Claims 41-47 have been added.
7. Claims 27, 34-35 and 37-40 have been cancelled.
8. Claims 1-26, 28-33, 36 and 41-47 are pending.
9. The objection to claim 22 is pending.
10. Claims 1-2, 4-16, 18-20, 28-33, 36 and 41-47 stand finally rejected under 35 U.S.C. 102(b) as being anticipated by Fortin (U.S. 5,528,753).
11. Claims 3 and 17 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Fortin (U.S. 5,528,753).
12. Claims 21-26 stand finally rejected under 35 U.S.C. 103(a) as being unpatentable over Fortin (U.S. 5,528,753), and further in view of Peek (U.S. 5,481,706).

Response to Amendment

Claim Objections

13. As previously pointed out in Paper no. 7, claim 22 is objected to because of the following informalities: claim 22 depends on itself. Claim 22 is interpreted as depending on independent claim 21. Appropriate correction is required.

Claim Rejections - 35 USC § 102

14. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

15. Claims 1-2, 4-16, 18-20, 28-33, 36 and 41-47 are rejected under 35 U.S.C. 102(b) as being anticipated by Fortin (U.S. 5,528,753).

Per Claim 1 (Amended):

The Fortin patent discloses:

- a computerized method for creating an instrumented executable file (“provide a system and method for efficiently instrumenting stripped object routines executing on a computer system” in column 3, lines 19-21)

- modifying an executable file having an original function with a user-supplied function

(“The Call to Target still points to address 202. After 202, however, control is passed to Entry Routine 210. Entry Routine 210 collects the information desired by the monitor and returns control to the target routine.” in column 4, lines 47-51)

- retaining access information of the original function, the access information enabling the user-supplied function to invoke the original function (“The Call to Target still points to address 202 ... returns control to the target routine.” in column 4, lines 47-51).

Per Claim 2:

The Fortin patent discloses:

- the user-supplied function is modified to invoke the original function using the retained access information of the original function (column 4, lines 49-53).

Per Claim 4:

The Fortin patent discloses:

- the user-supplied function is not exported during compilation (column 5, lines 20-24).

Per Claim 5:

The Fortin patent discloses:

- the original function and the user-supplied function have identical prototypes (column 4, lines 62-65).

Per Claim 6:

The Fortin patent discloses:

- the user-supplied function is stored in a module that is separate from the executable file (column 5, lines 20-24).

Per Claim 7 (Amended):

The Fortin patent discloses:

- modifying the executable file is performed using user-specified set points (column 4, lines 54-65).

Per Claim 8:

The Fortin patent discloses:

- determining whether the original function implements the thiscall calling convention, and when the determination is positive, adding instructions to the executable file to perform pushing the register that holds the “this” pointer onto the stack from the invoked original

function site when the determining indicates that the function implements a thiscall calling convention (column 5, lines 48-51 and lines 55-57, and column 6, lines 48-58)

- swapping the return value of the invoking original function on the stack and the register that holds the “this” pointer value on the stack when the determining indicates that the function implements a thiscall calling convention (column 6, lines 59-67).

Per Claim 9 (Amended):

The Fortin patent discloses:

- modifying the executable file further comprises enabling the user-supplied function to invoke the original function in the executable file (column 4, lines 49-53).

Per Claim 10:

The Fortin patent discloses:

- adding a jump in the user-supplied function to a function that retrieves the address of the original function (column 6, lines 6-18)

- adding a jump in the user-supplied function that invokes the original function using the address of the original function (column 6, lines 1-5 and lines 35-39).

Per Claim 11:

The Fortin patent discloses:

- enabling the user-supplied function to alter behavior (column 4, lines 47-53).**

Per Claim 12:

The Fortin patent discloses:

- enabling the user-supplied function to alter behavior is performed in response to data (column 4, lines 47-53).**

Per Claim 13:

The Fortin patent discloses:

- the data is retrieved from an initialization file (“libinstr.a” in Fig. 7 and column 5, lines 20-24).**

Per Claim 14:

The Fortin patent discloses:

- saving the address of an original function in a threaded local storage variable (column 6, lines 1-5)**

- creating an entry in a function lookup table associating the address of the original function with the name of the original function, wherein the function lookup table is in the instrumented executable file (column 5, lines 46-51 and lines 64-67).

Per Claim 15 (Amended):

The Fortin patent discloses:

- a computerized method for executing an instrumented executable file (“provide a system and method for efficiently instrumenting stripped object routines executing on a computer system” in column 3, lines 19-21)

- modifying the instrumented executable file having an original function with a user-supplied function, the user-supplied function having a jump to the original function (“The Call to Target still points to address 202. After 202, however, control is passed to Entry Routine 210. Entry Routine 210 collects the information desired by the monitor and returns control to the target routine. ... The stored target routine first instruction is executed in the ‘Return to Target’ section and control branches to the second instruction of the target routine 714.” in column 4, lines 47-51 and column 6, lines 56-58)

- saving the address of the original function in a threaded local storage variable (“The ‘Return to Target Routine’ section 604 contains the ‘saved’ first instruction of the target routine

and direct branch to the address of the second instruction of the target routine.” in column 6, lines 1-5)

- invoking the user-supplied function using the address (“Processing begins with a call to the target routine 702. This call will encounter the branch first instruction and immediately branch to the Entry section 704 … that in turn calls 708 user supplied Entry routine” in column 6, lines 46-51).

Per Claim 16:

The Fortin patent discloses:

- creating a master lookup table at initialization wherein the master lookup table associates the base address of the instrumented executable file to the address of a function lookup table in the instrumented executable file (column 5, lines 47-51 and lines 55-67).

Per Claim 18:

The Fortin patent discloses:

- wherein original function is embedded in the instrumented executable file, and wherein the saving and the invoking is performed by the original function (column 6, lines 46-51).

Per Claim 19:

The Fortin patent discloses:

- **invoking the original function from within the user-supplied function using the threaded local storage variable** (column 6, lines 56-58).

Per Claim 20:

The Fortin patent discloses:

- **pushing the register that holds the “this” pointer onto the stack from the invoked original function site when the determining indicates that the function implements a thiscall calling convention** (column 5, lines 48-51 and lines 55-57, and column 6, lines 48-58)
- **swapping the return value of the invoking original function on the stack and the register that holds the “this” pointer value on the stack when the determining indicates that the function implements a thiscall calling convention** (column 6, lines 59-67).

Per Claim 27:

The Fortin patent discloses:

- **a computerized method for publishing a function** (“provide a system and method for efficiently instrumenting stripped object routines executing on a computer system” in column 3, lines 19-21)

- adding an entry describing the function in a function lookup table in a machine-readable executable file (“insertion of the demultiplexor entries 510. A demultiplexor entry (demux-entry) is provided for each target routine.” in column 5, lines 46-51 and lines 64-67).

Per Claim 28 (Amended):

This is a system version of the claimed method discussed above, claim 1, wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, this claim is also anticipated by Fortin.

Per Claim 29 (Amended):

This is a system version of the claimed method discussed above (claims 1 and 2), wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above, including “the original function having an identity comprising a name and a parameter prototype” (column 4, lines 62-65 and column 5, lines 64-67). Thus, accordingly, this claim is also anticipated by Fortin.

Per Claim 30 (Amended):

The Fortin patent discloses:

- a computerized system (“provide a system and method for efficiently instrumenting stripped object routines executing on a computer system” in column 3, lines 19-21)

- **an executable file having a jump to an original function, the original function having an identity comprising a name and a parameter prototype** (“The stored target routine first instruction is executed in the “Return to Target” section and control branches to the second instruction of the target routine 714.” in column 5, lines 64-67 to column 6, lines 1-5 and lines 56-58)
- **a first software component having a user-supplied function that includes a jump to the original function** (“The Entry routine saves … and calls 706 the common Entry code that in turn calls 708 user supplied Entry routine … The stored target routine first instruction is executed in the “Return to Target” section and control branches to the second instruction of the target routine 714.” in column 5, lines 64-67 to column 6, lines 1-5 and lines 48-58)
- **a second software component for receiving the identity of the original function, receiving the identity of the user-supplied function, instrumenting the executable file by modifying the identity of the original function with the identity of the user-supplied function** (“The routines also allow the function of a target routine to be fully replaced such that newly provided code will be executed instead of the base code in the routine being monitored.” in column 4, lines 62-65)
- **storing the original function address in the executable file in association with the name of the original instrumented function** (“insertion of the demultiplexor entries 510. A

demultiplexor entry (demux-entry) is provided for each target routine ... The data section 602 contains the addresses at which the target routine and the target routine's symbolic name reside" in column 5, lines 46-51 and lines 64-67).

Per Claim 31 (Amended):

The Fortin patent discloses:

- **a computerized system** ("provide a system and method for efficiently instrumenting stripped object routines executing on a computer system" in column 3, lines 19-21)
- **a first module of machine-readable code comprising a call to an original function, the call being directed to a user-supplied function** ("The Call to Target still points to address 202. After 202, however, control is passed to Entry Routine 210. Entry Routine 210 collects the information desired by the monitor and returns control to the target routine." in column 4, lines 47-51)
- **a first data structure associating the identity of the original function with the location of the original function** ("A demultiplexor entry (demux-entry) is provided for each target routine ... The data section 602 contains the addresses at which the target routine and the target routine's symbolic name reside" in column 5, lines 46-51 and lines 64-67)

- a second module comprising the user-supplied function, linked to the first module and a jump to the original function (“The stored target routine first instruction is executed in the ‘Return to Target’ section and control branches to the second instruction of the target routine 714.” in column 6, lines 56-58).

Per Claim 32:

The Fortin patent discloses:

- the first data structure comprises a function lookup table readily available for verifying that the threaded local storage variable contains the correct original instrumented function address (column 5, lines 64-67 to column 6, lines 1-5)

- the second module comprises a dynamic linked library (column 5, lines 20-24).

Per Claim 33:

The Fortin patent discloses:

- a second data structure associating the location of the first data structure with the location of the first module (column 5, lines 47-51 and lines 55-67).

Per Claim 34:

The Fortin patent discloses:

- a third module of machine-readable code comprising a second instrumented function call to a second replacement function (column 4, lines 47-51 and column 5, lines 48-49, one or more instrumented functions)
- a second data structure associating the identity of the second instrumented function with the location of the second instrumented function (column 5, lines 55-67)
- a fourth module comprising the second replacement function, having a jump to the second original function (column 6, lines 56-58).

Per Claim 35:

The Fortin patent discloses:

- a third module of machine-readable code comprising a second instrumented function jump to a second replacement function (column 6, lines 59-67 and column 5, lines 48-49, one or more instrumented functions)
- a second data structure associating the identity of the second instrumented function with the location of the second instrumented function (column 5, lines 55-67)

- the second module further comprises the second replacement function, having a jump to the second original function (column 6, lines 56-58).

Per Claim 36 (Amended):

This is a computer-readable medium version of the claimed method discussed above, claim 1, wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, this claim is also anticipated by Fortin.

Per Claim 37:

This is a computer-readable medium version of the claimed system discussed above (claims 31 and 33), wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, this claim is also anticipated by Fortin.

Per Claims 38-39:

This is a computer-readable medium version of the claimed system discussed above (claims 31-32, respectively), wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, this claim is also anticipated by Fortin.

Per Claim 40:

This is a computer-readable medium version of the claimed system discussed above, claim 32, wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, this claim is also anticipated by Fortin.

Per Claims 41 (New) & 43 (New):

These are another versions of the claimed method discussed above, claim 1, wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above, including “replacing the access to the original function with an access to a user-supplied function” (column 4, lines 47-51). Thus, accordingly, these claims are also anticipated by Fortin.

Per Claim 42 (New):

This is another version of the claimed method discussed above, claim 2, wherein all claim limitations also have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, this claim is also anticipated by Fortin.

Per Claim 44 (New):

The Fortin patent discloses:

- wherein replacing the access to the original function with the access to the user-supplied function is performed by modifying set points stored in a computer-readable medium separate from the executable file (column 4, lines 54-65).

Per Claim 45 (New):

The Fortin patent discloses:

- **wherein retaining access information associated with the original function includes saving the address of the original function** (column 6, lines 1-5).

Per Claim 46 (New):

The Fortin patent discloses:

- **wherein retaining access information associated with the original function includes associating the name of the original function with the address of original function using a function lookup table** (column 5, lines 46-51 and lines 64-67).

Per Claim 47 (New):

The Fortin patent discloses:

- **further comprising invoking the original function using the function lookup table** (column 5, lines 46-51 and lines 64-67).

Claim Rejections - 35 USC § 103

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

17. Claims 3 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fortin (U.S. 5,528,753).

Per Claim 3 (Amended):

The rejection of claim 1 is incorporated, and Fortin further teaches that the user-supplied function is in a library (column 5, lines 20-24). Fortin does not explicitly teach that the user-supplied function is in a dynamic link library. Official Notice is taken that it was a common practice to have functions in a dynamic link library.

It would have been obvious to one having ordinary skill in the computer art at the time of the invention was made to modify the method disclosed by Fortin to include that the user-supplied function is in a dynamic link library using the teaching of common practice. The modification would be obvious because one of ordinary skill in the art would be motivated to modify functions without recompiling the program.

Per Claim 17:

The rejection of claim 15 is incorporated, and Fortin teaches that the saving and the invoking are performed by the original function, and the original function being located in the instrumented executable file (column 6, lines 44-51). Fortin does not explicitly teach that the original function is in a dynamic link library and a stub function of the original function being located in the instrumented executable file. Official Notice is taken that it was a common

practice to include a function in a dynamic link library at the time the instant invention was made.

It would have been obvious to one having ordinary skill in the computer art at the time of the invention was made to modify the method disclosed by Fortin to include a function in a dynamic link library using the teaching of common practice. The modification would be obvious because one of ordinary skill in the art would be motivated to eliminate recompilation of a file, save memory space, and one would also want to provide the ease of modifying functions.

18. Claims 21-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fortin (U.S. 5,528,753), and further in view of Peek (U.S. 5,481,706).

Per Claims 21 (Amended) & 22 (as best understood):

Fortin teaches a computerized method for instrumenting an original function in an executable file for testing by callers of the original function (“provide a system and method for efficiently instrumenting stripped object routines executing on a computer system” in column 3, lines 19-21), the original function comprises an instruction that saves the addresss of the original function to a threaded local storage variable (“The ‘Return to Target Routine’ section 604 contains the ‘saved’ first instruction of the target routine and direct branch to the address of the second instruction of the target routine.” in column 6, lines 1-5), replaces an access to the original function with an access to the user-supplied function (“Processing begins with a call to the target routine 702. This call will encounter the branch first instruction and immediately branch to the Entry section 704 ... that in turn calls 708 user supplied Entry routine” in column

6, lines 46-51), adding an entry in a function lookup table of the original function (“insertion of the demultiplexor entries 510. A demultiplexor entry (demux-entry) is provided for each target routine.” in column 5, lines 46-51 and lines 64-67), determining if the prototype of the original function is correctly specified and indicating an error when the determining indicates an incorrectly specified prototype of the original function (“The routines also allow the function of a target routine to be fully replaced such that newly provided code will be executed instead of the base code in the routine being monitored.” in column 4, lines 62-65, inherently determining prototype). Fortin does not explicitly teach an imported function, a stub function for the imported function, and adding a wrapper of the imported function to an import data block. Official Notice is taken that it was a common practice to include a stub function for an imported function at the time the instant invention was made. Peek teaches adding a wrapper of a function to a data block (“providing a “wrapper” for the particular function ... a wrapper will be seen to be an individual piece of code placed around a corresponding individual function call or routine” in column 6, lines 34-37 and 43-50).

It would have been obvious to one having ordinary skill in the computer art at the time of the invention was made to modify the method disclosed by Fortin to include a stub function for an imported function and adding a wrapper of the imported function to an import data block using the teaching of common practice and Peek. The modification would be obvious because one of ordinary skill in the art would be motivated to eliminate recompilation of a file and to encapsulate data.

Per Claims 23 (Amended) & 24-25:

Fortin teaches a computerized method for instrumenting an original function in an executable file for testing by callers of the original function (“provide a system and method for efficiently instrumenting stripped object routines executing on a computer system” in column 3, lines 19-21), modifying an original function with a user-supplied function (“The Call to Target still points to address 202. After 202, however, control is passed to Entry Routine 210. Entry Routine 210 collects the information desired by the monitor and returns control to the target routine.” in column 4, lines 47-51), adding an entry in a function lookup table of the address of the original function (“insertion of the demultiplexor entries 510. A demultiplexor entry (demux-entry) is provided for each target routine.” in column 5, lines 46-51 and lines 64-67), redirecting is accomplished by an instruction that causes a jump to the user-supplied function (“Processing begins with a call to the target routine 702. This call will encounter the branch first instruction and immediately branch to the Entry section 704 … that in turn calls 708 user supplied Entry routine” in column 6, lines 46-51), determining whether the prototype of the original function is correctly specified, and indicating an error when the determining whether the prototype of the original function is correctly specified indicates an incorrectly specified prototype of the original function (“The routines also allow the function of a target routine to be fully replaced such that newly provided code will be executed instead of the base code in the routine being monitored.” in column 4, lines 62-65, inherently determining prototype). Fortin does not explicitly teach an embedded function and modifying an embedded function using a wrapper. Official Notice is taken that it was a common practice to include an embedded function at the time the instant invention was made. Peek teaches modifying a function using a wrapper (“providing a “wrapper” for the particular function … a wrapper will be seen to be an individual

piece of code placed around a corresponding individual function call or routine" in column 6, lines 34-37 and 43-50).

It would have been obvious to one having ordinary skill in the computer art at the time of the invention was made to modify the method disclosed by Fortin to include an embedded function and modifying an embedded function using a wrapper using the teaching of common practice and Peek. The modification would be obvious because one of ordinary skill in the art would be motivated to encapsulate functions.

Per Claim 26:

Fortin further teaches that the function lookup table is in the executable file (column 5, lines 46-51).

Response to Arguments

19. Applicant's arguments with respect to claims 1-26,28-33,36 and 41-47 have been fully considered but they are not persuasive.

In the remarks, the applicant argues that:

a) The procedure in Fortin cited by the Office Action monitors a target routine by passing control to other routines within the target routine. However, unlike the method recited in Claim 1 of the applicants' application, the procedure in Fortin does not include modifying an executable file having an original function with a user-supplied function. Instead, the procedure described in Fortin simply passes control of a program to some monitoring routines and the target routine always executes. The procedure in Fortin also fails to describe retaining information about the

target routine subsequent to replacing the routine. Thus, the Fortin procedure is different and, more significantly, teaches away from the method in Claim 1.

Examiner's response:

a) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 1. Fortin shows each and every limitation in claim 1. Fortin teaches modifying an executable file having an original function with a user-supplied function (column 4, lines 47-51, the executable file is being modified by the Entry routine) and retaining access information of the original function, the access information enabling the user-supplied function to invoke the original function (column 4, lines 47-51, control is returned to the target routine). In addition, see the rejection above in paragraph 15 for rejection to claim 1.

In the remarks, the applicant argues that:

b) Moreover, the method recited in Claim 1 can be applied to executable file that includes functions in a dynamic link library, unlike the procedure in Fortin. Throughout the specification, the present invention is described in detail as to how it can be applied to an executable file having an original function and a user-supplied function that are linked to it. For example, the original function and the user-supplied function may be placed in a dynamic link library. (See applicant's application, page 4 lines 4-14 and page 14, lines 1-3).

Examiner's response:

b) In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., the method recited in Claim 1 can be applied to executable file that includes functions in a dynamic link library) are not recited in the rejected claim 1. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

In the remarks, the applicant argues that:

c) Dependent Claim 3 particularly claims this aspect of the invention and recites that "the user-supplied function is in a dynamic link library." In contrast, Fortin particularly disclaims linked objects. The invention in Fortin is only intended for software programs that are stripped of any linkable information. (Fortin, column 3, lines 1-2). In particular, because stripped objects do not have a symbol table, the instrumentation library described in Fortin cannot simply be linked to the objects. (Fortin, column 5, lines 27-29). Thus, Fortin also fails to describe the method recited in Claims 1 and 3, which works with linked functions.

Examiner's response:

c) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 3. Dynamic link library (DLL) is a data structure that stores objects, even stripped objects; furthermore, DLL has its own mechanism for links. In addition, see the rejection above in paragraph 17 for rejection to claim 3.

In the remarks, the applicant argues that:

d) The Office Action rejected independent Claim 15 by essentially citing the same procedure in Fortin that the Office Action used to reject Claim 1. Independent Claim 15, as amended, is directed to a computerized method for executing an instrumented executable file. The method includes modifying the instrumented executable file having an original function with a user-supplied function. The user-supplied function has a jump to the original function. The method also includes saving the address of the original function in a threaded local storage variable and invoking the user-supplied function using the address. None of these steps is described in Fortin.

As mentioned above in the discussion regarding Claim 1, Fortin merely describes a procedure for monitoring a target routine by passing control to other routines within the target routine. Fortin does not disclose modifying an executable file with a user-supplied function where, as described in the specification, the executable file is modified to replace, substitute or instrument the access to the original function with an access to the user-supplied function. Furthermore, Fortin fails to mention the use of threaded local storage (TLS) variable and saving the address of the original function as a TLS. Thus, applicants respectfully submit that the invention recited in independent Claim 15 is not anticipated or rendered obvious by Fortin and is allowable. Dependent Claims 16-20 are also allowable for at least the same reasons.

Examiner's response:

d) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 15. The Examiner has already addressed the applicant's argument in

Examiner's Response (a) above. Furthermore, Fortin teaches saving the address of the original function in a threaded local storage variable (column 6, lines 1-5). In addition, see the rejection above in paragraph 15 for rejection to claim 15.

In the remarks, the applicant argues that:

e) The Office Action also rejected Claim 16, which includes creating a master lookup table at initialization wherein the master lookup table associates the base address of the instrumented executable file to the address of a function lookup table in the instrumented executable file. As described in the specification, the master lookup table is used to locate the function lookup table and the function lookup table is used to match the name of the original function to the function's address. (Applicants' application, page 18). Neither of the two tables is described by Fortin.

The section cited by the Office Action to reject Claim 16 merely describes a demultiplexor entry used for monitoring a target routine. In particular, the demultiplexor entry serves to direct an instrumentation call to the appropriate common and user specific entry and exit routines. (Fortin, column 5, lines 49-51). However, the demultiplexor entry is not a function table recited in Claim 16 that matches the name of an original function to the function's address. Thus, applicants respectfully submit that dependent Claim 16 is allowable for at least this additional reason.

Examiner's response:

e) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 16. As previously pointed out in Paper no. 7, the demultiplexor entry

contains a data section that matches the name of an original function to the function's address (column 5, lines 47-51 and lines 55-67, the demultiplexor is interpreted as a function table).

In the remarks, the applicant argues that:

f) The Office Action also rejected Claim 19, which includes invoking the original function from within the user-supplied function using the threaded local storage variable. As mentioned above, Fortin fails to disclose the use of threaded local storage variables. Also, Fortin merely describes monitoring a target routine in a stripped object by passing control to monitoring routines while the target routine is executing. Fortin does not describe invoking an original function from within a user-supplied function, as recited in Claim 19. Applicants respectfully submit that dependent Claim 19 is allowable for at least this additional reason.

Examiner's response:

f) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 19. The Examiner has already addressed the applicant's argument in Examiner's Response (d) above. Furthermore, Fortin teaches invoking the original function from within the user-supplied function (column 6, lines 56-58). In addition, see the rejection above in paragraph 15 for rejection to claim 19.

In the remarks, the applicant argues that:

Art Unit: 2124

g) The Office Action rejected Claims 28-29 and 36 using the same basis as those used for rejecting Claim 1. Accordingly, applicants respectfully submit that Claims 28-29 and 36, as amended, are allowable for at least the same reasons stated above regarding Claim 1.

Examiner's response:

g) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claims 28-29 and 36. The Examiner has already addressed the applicant's argument in Examiner's Response (a) above regarding claim 1. In addition, see the rejection above in paragraph 15 for rejection to claims 28-29 and 36.

In the remarks, the applicant argues that:

h) Claim 30, as amended, ... As discussed above, Fortin merely describes a system and method for monitoring a target routine in a stripped object by passing control within the target routine to some monitoring routines. Fortin does not describe a component for "instrumenting the executable file by modifying the identity of the original function with the identity of the user-supplied function" or "storing the original function address in the executable file in association with, the name of the original instrumented function." Thus, applicants respectfully submit that Claim 30 is not anticipated by Fortin and is allowable.

Examiner's response:

h) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 30. The Examiner has already addressed the applicant's argument in

Examiner's Response (a) above. In addition, see the rejection above in paragraph 15 for rejection to claim 30.

In the remarks, the applicant argues that:

i) Claim 31, as amended, ... As discussed above, Fortin only discloses a system and method for monitoring a stripped object. Nothing in Fortin describes a system that includes a first module that is linked to a second module having a user-supplied function, as recited in Claim 31. Thus, applicants respectfully submit that Claim 31 is not anticipated by Fortin and is allowable. Dependent Claims 32-33 are allowable for at least the same reasons.

Examiner's response:

i) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 31. The Examiner has already addressed the applicant's argument in Examiner's Response (a) above. In addition, see the rejection above in paragraph 15 for rejection to claim 31.

In the remarks, the applicant argues that:

j) Claim 17 were rejected under 35 U.S.C. 103(a) as being unpatentable over Fortin and based on an assertion that a function in a dynamic link library was common practice. Claim 17 adds to independent Claim 15 that the original function is in a dynamic link library. As discussed in Section D above, the procedure in Fortin is only for monitoring stripped objects, which means objects without any linkable information. Even if the use of a dynamic lank library is known,

Fortin expressly excludes the use of linkable information and, thus, teaches away from it. Any attempt to combine Fortin and the use of a dynamic link library with merely an assertion is inappropriate and without merit. According, applicants respectfully submit that Claim 17 is not rendered obvious by Fortin and is allowable.

Examiner's response:

j) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 17. The Examiner has already addressed the applicant's argument in Examiner's Response (c) above. In addition, see the rejection above in paragraph 17 for rejection to claim 17.

In the remarks, the applicant argues that:

k) Independent Claim 21 were rejected under 35 U.S.C. 103(a) as being unpatentable over Fortin and further in view of Peek, USPN 5,481,706. As amended, Claim 21 includes ... As discussed above, Fortin fails to disclose modifying an original function with a user-supplied function. Fortin also fails to disclose a function lookup table, which is described in the applicants' specification as used for matching the name of the original function to the function's address. These deficiencies are not remedied by Peek.

The Office Action asserts that including a stub function is common practice and that Peek teaches adding a wrapper of a function to a data block. However, even if true, these assertions still do not describe the subject matter recited in Claim 21. Thus, applicants respectfully submit

that independent Claim 21 is not rendered obvious by any combination of Fortin and Peek and is allowable. Dependent Claims 22-26 are also allowable for at least the same reason.

Examiner's response:

k) Examiner disagrees with applicant's assertion that Fortin fails to disclose the claimed limitations recited in claim 21. The Examiner has already addressed the applicant's argument in Examiner's Responses (a) and (e) above. In addition, see the rejection above in paragraph 18 for rejection to claims 21-26.

Conclusion

20. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Art Unit: 2124

21. Any inquiry concerning this communication from the examiner should be directed to Qamrun Nahar whose telephone number is (703) 305-7699. The examiner can normally be reached on Mondays through Thursdays from 9:00 AM to 6:30 PM. The examiner can also be reached on alternate Fridays.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki, can be reached on (703) 305-9662. The fax phone number for the organization where this application or processing is assigned is (703) 746-7239.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

QN
June 30, 2003

Kakali Chaki
KAKALI CHAKI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100